

daegu apartment

Price Prediction | Machine learning



ANDI SETIANTO

andisetianto123@gmail.com

Road map



Background?

Approach?

Analytics?

Background

Urgensi / Problem
Tentang Data

Approach

EDA
Data Cleaning / Manipulation
Asumsi
Splitting
Cross Validation & Pipe Line
Predict Test Set & Benchmark
Best Model

Analytics

Conclusion
Recommendation
Bonus! SHAPASH



1

Background



problem

Sejumlah 2000 properti di Daegu Tidak terjual
(statista, 2021)

Pemilik unit susah untuk bertemu

Perlu kemudahan dalam menentukan harga yang wajar

Permintaan akan properti meningkat
(statista, 2021)



permintaan properti

240000 unit

Properti belum terjual

2000 unit

populasi

2.4 juta

Data

source

Kaggle or

<https://drive.google.com/drive/folders/1fmkyfjrzuJNaH02sXhp5vUxqum9bHOFx>

explanation

- Hallway Type: Apartment type
- TimeToSubway: Time needed to the nearest subway station
- SubwayStation: The name of the nearest subway station
- N_FacilitiesNearBy(ETC): The number of facilities nearby
- N_FacilitiesNearBy(PublicOffice): The number of public office facilities nearby
- N_SchoolNearBy(University): The number of universities nearby
- N_Parkinglot(Basement): The number of the parking lot
- YearBuilt: The year the apartment was built
- N_FacilitiesInApt: Number of facilities in the apartment
- Size(sqft): The apartment size (in square feet)
- SalePrice: The apartment price (Won)

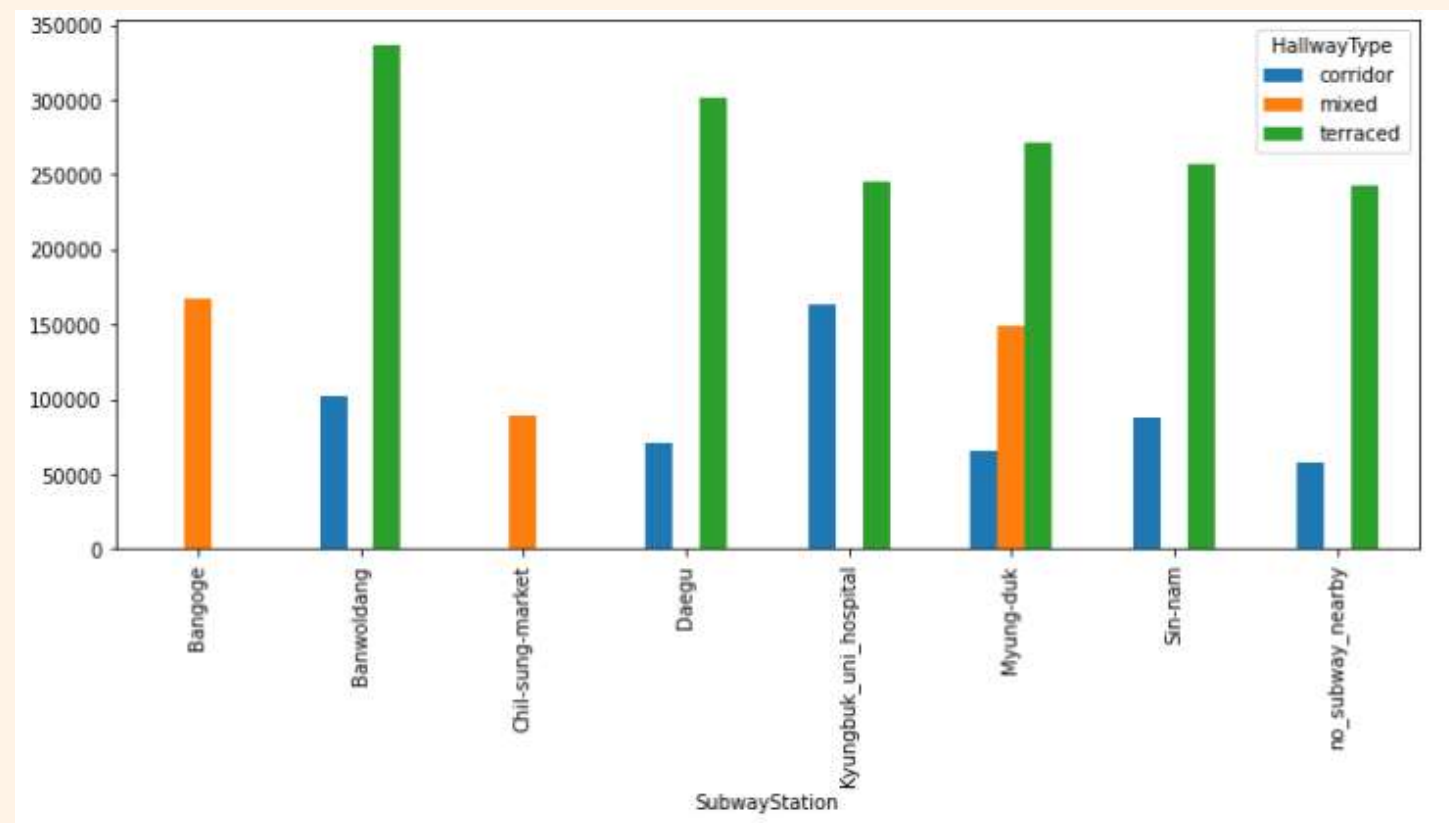
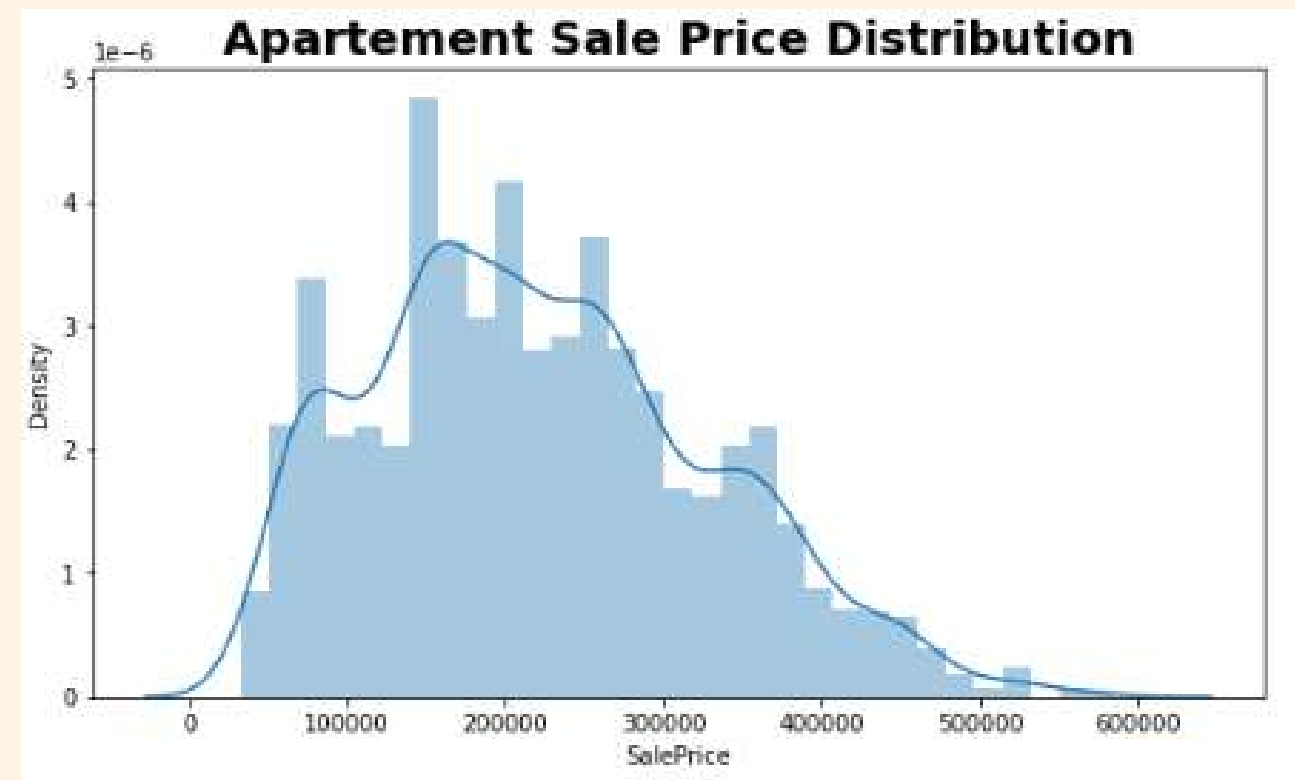
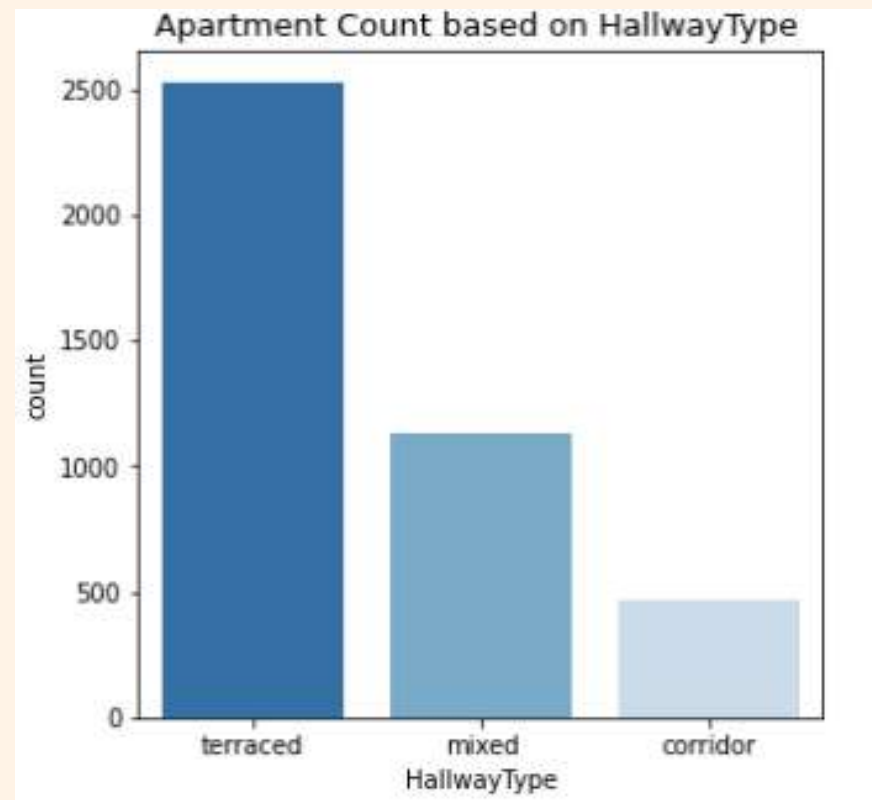


2

Approach



EDA



Banwoldang:

Satu satunya stasiun yang terhubung dengan Daegu Metro dan banyak tempat nongkrong

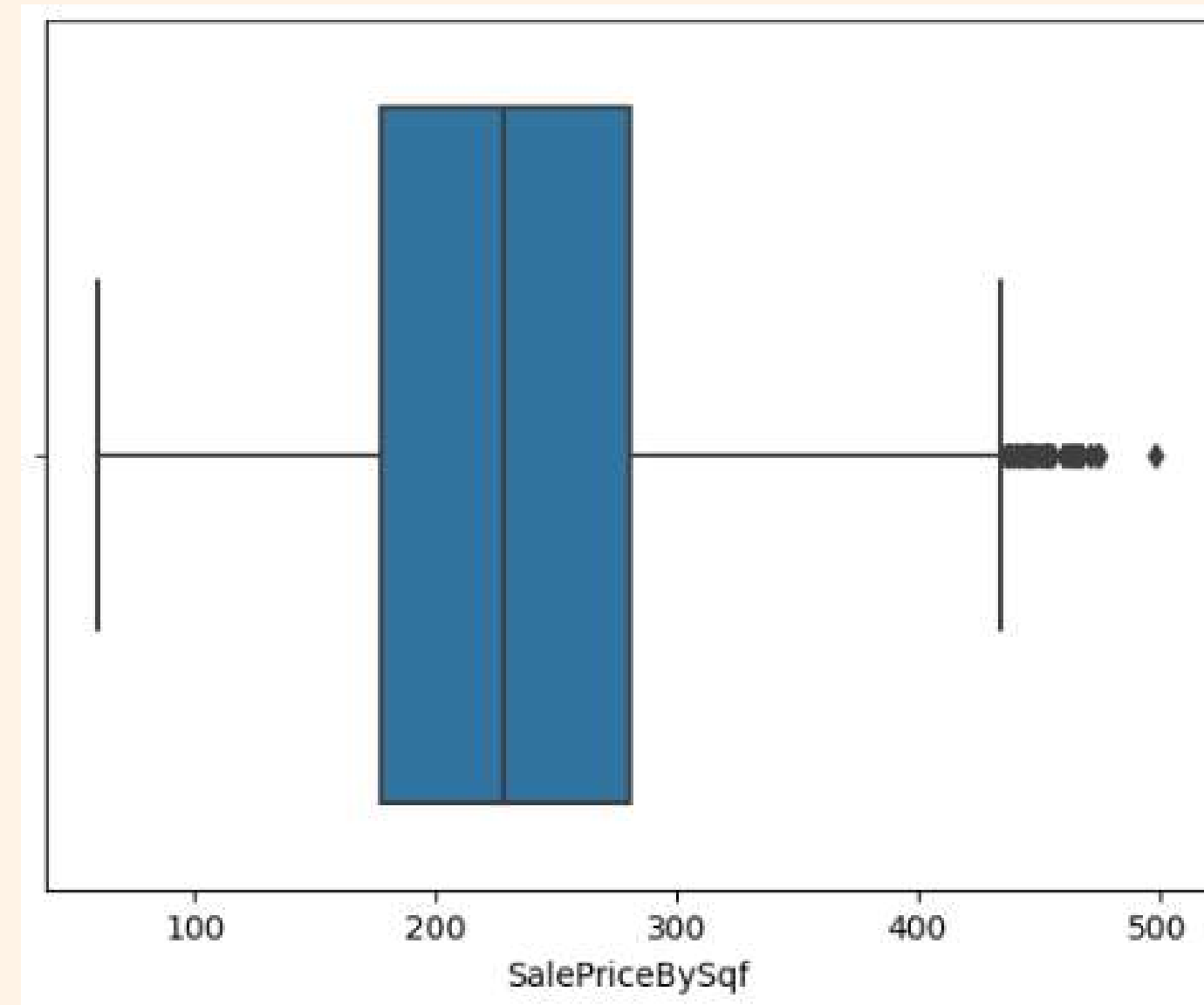


Data Cleaning /Manipulation

Missing Values

Outlier

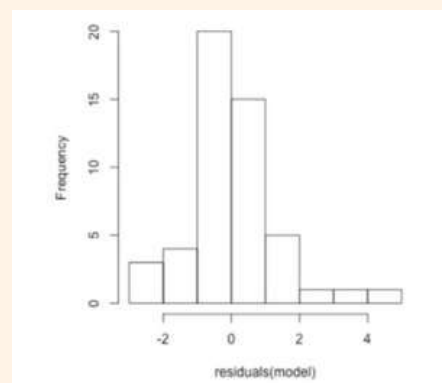
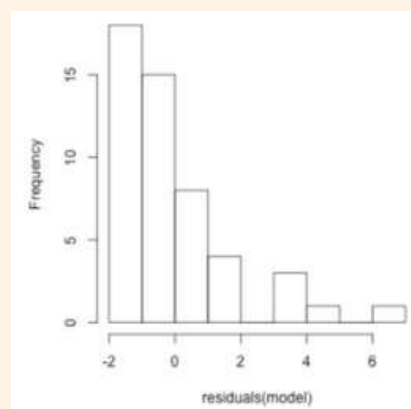
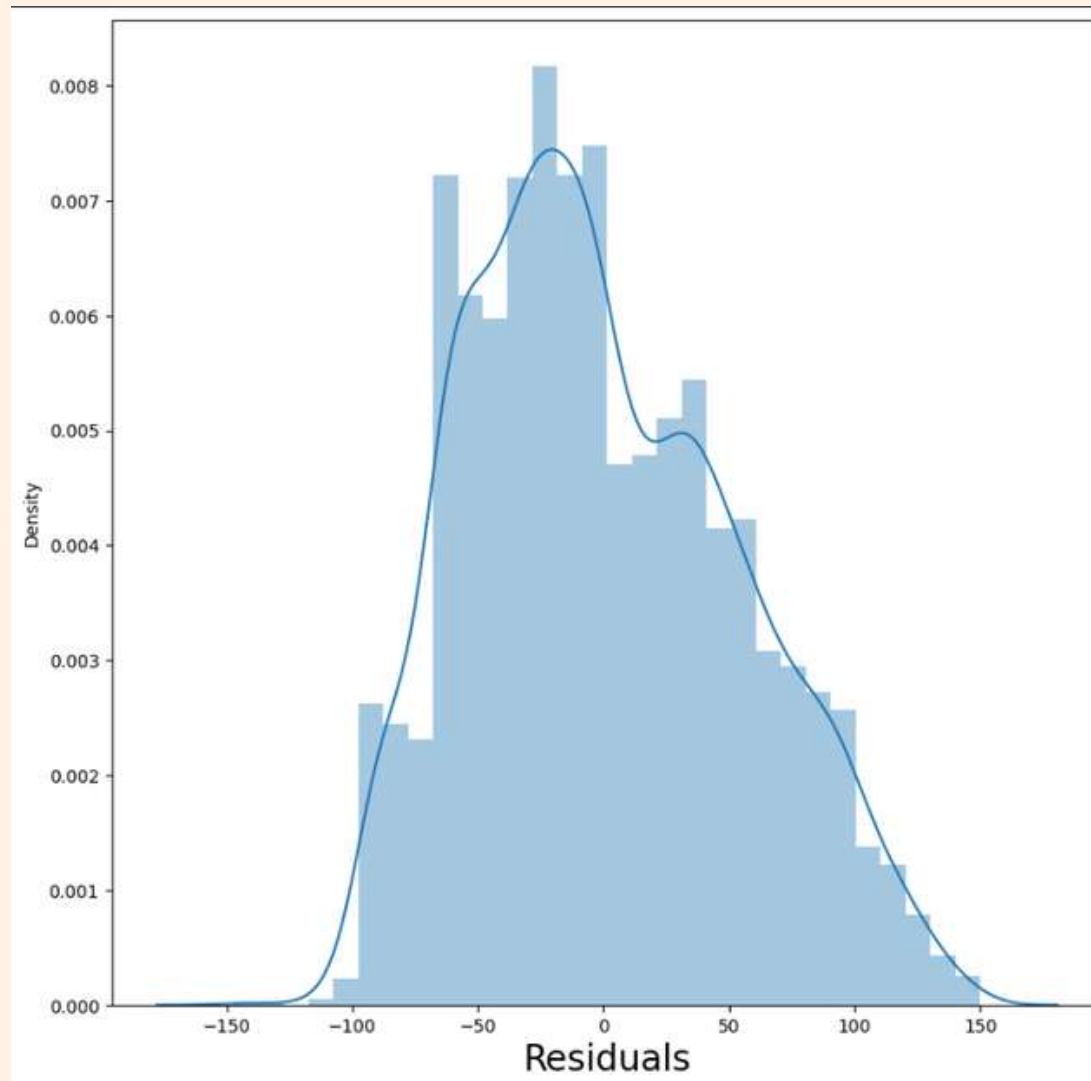
Duplication



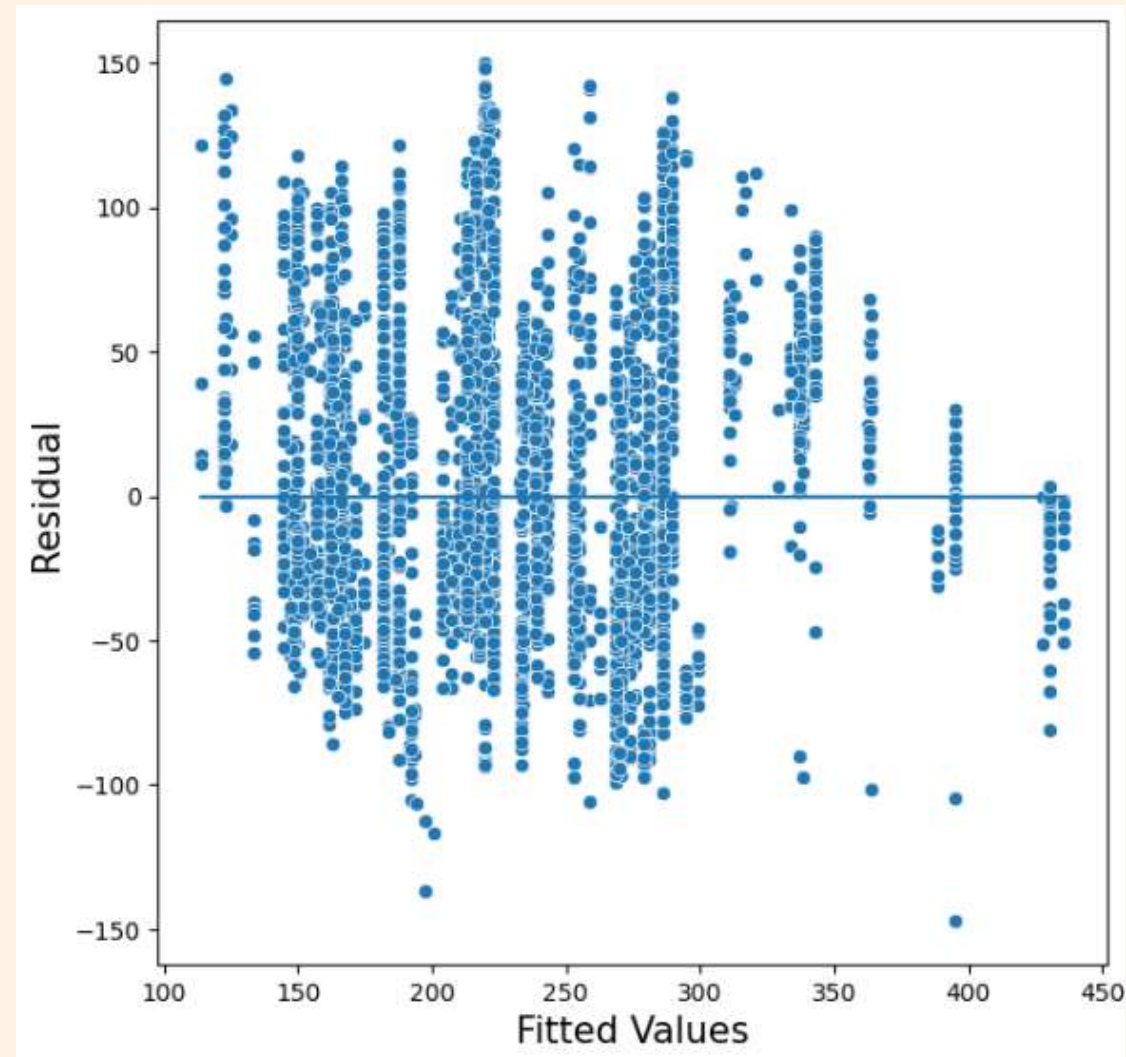
```
IQR: 146398.0  
limit_bawah: -74845.0  
limit_atas: 510747.0
```


Cek Asumsi

Distribusi



Gauss Markov

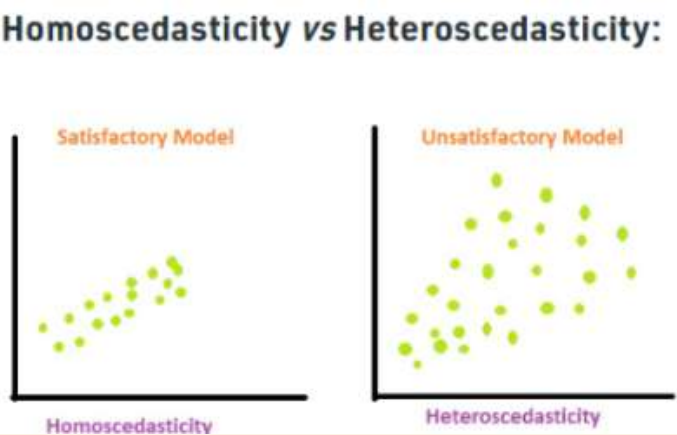
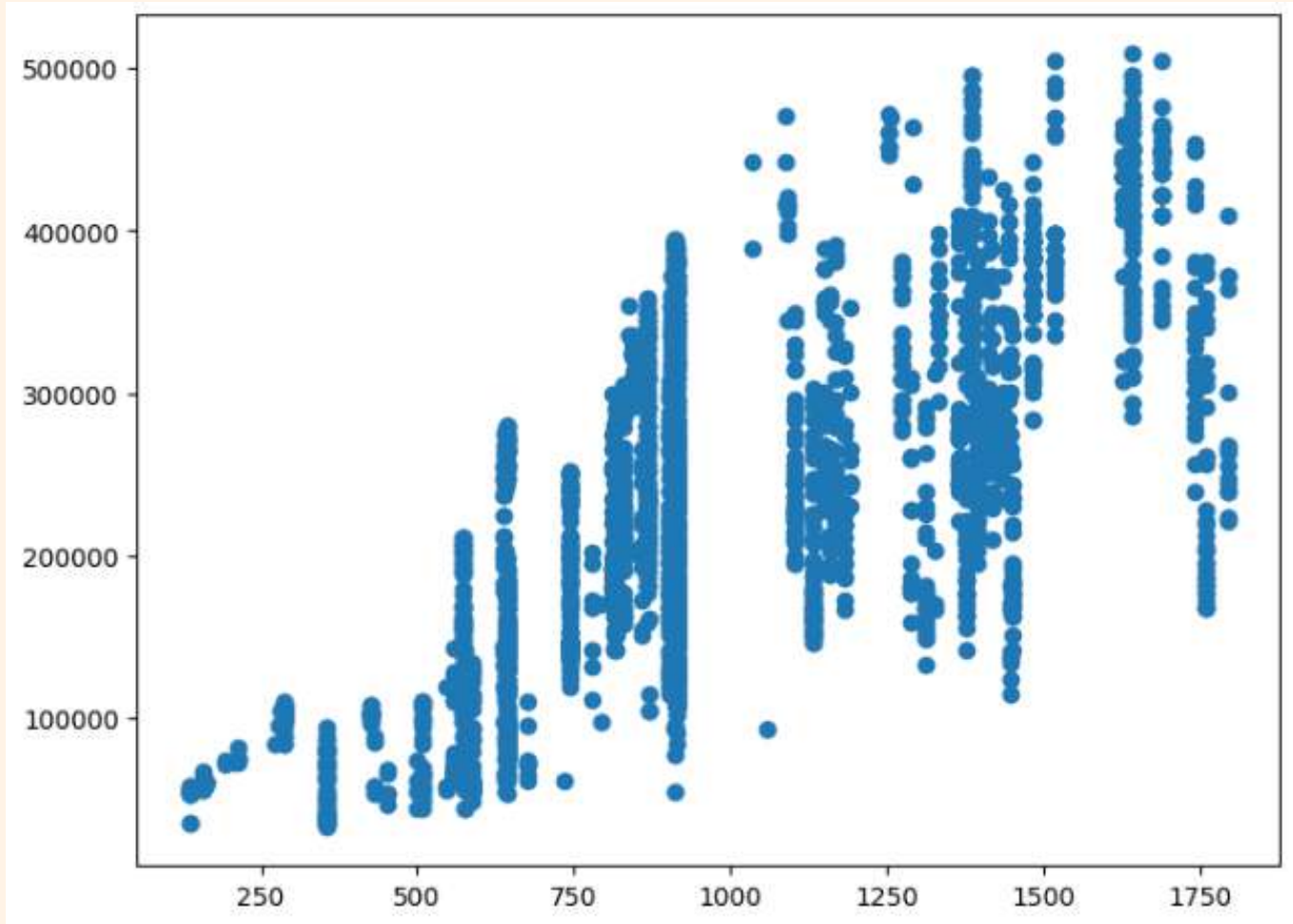


Asumsi dalam linear regresi, sebaiknya memenuhi kriteria berikut:

1. Residual harus indendent antara satu dengan lain
2. Residual harus memiliki variance yang constant
3. Expected value harus / mean dari residuals harus mendekati 0
4. Residual seharusnya terdistribusi normal

Cek Asumsi

Homoskedasity



Multicollinearity

	variables	VIF	Acceptable
0	N_FacilitiesNearBy(ETC)	28.202046	Highly Correlated
1	N_FacilitiesNearBy(PublicOffice)	28.221791	Highly Correlated
2	N_SchoolNearBy(University)	31.325676	Highly Correlated
3	ageBuilding	20.852526	Highly Correlated
4	N_FacilitiesInApt	18.343496	Highly Correlated
5	Size(sqf)	14.123662	Highly Correlated
6	TimeToSubway_10min~15min	3.923903	Not Correlated / Moderately
7	TimeToSubway_15min~20min	5.066479	Highly Correlated
8	TimeToSubway_5min~10min	6.938167	Highly Correlated
9	TimeToSubway_no_bus_stop_nearby	inf	Highly Correlated
10	TimeToSubway_no_bus_stop_nearby	inf	Highly Correlated
11	SubwayStation_Banwoldang	5.749252	Highly Correlated
12	SubwayStation_Chil-sung-market	2.829497	Not Correlated / Moderately
13	SubwayStation_Daegu	1.564159	Not Correlated / Moderately
14	SubwayStation_Kyungbuk_uni_hospital	18.682945	Highly Correlated
15	SubwayStation_Myung-duk	12.118355	Highly Correlated
16	SubwayStation_Sin-nam	5.793792	Highly Correlated
17	SubwayStation_no_subway_nearby	5.424816	Highly Correlated

Berdasarkan informasi dari <https://towardsdatascience.com/how-to-remove-multicollinearity-using-python-4da8d9d8abb2> ,untuk mengurangi masalah multicollinearity, dapat dilakukan dengan penghapusan fitur

Solusi:

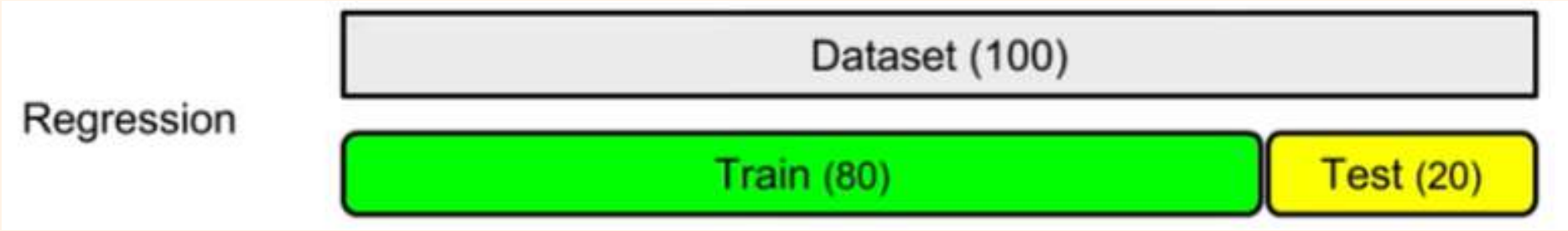
Menyingkirkan beberapa feature dengan VIF yang besar dan secara fungsi sudah terwakili oleh feature lainnya.

alasan remove:

Saya menyingkirkan Hallway Type Karena dianggap sudah bagian dari fasilitas / N_FacilitiesInApt.



Splitting & Encode



Target: SalePriceBySqm

One Hot Encoding

Color	One-hot encoding		
Red	d1	d2	d3
Green	1	0	0
Blue	0	1	0
	0	0	1

****One-Hot encoding**** adalah salah satu metode encoding untuk merepresentasikan data bertipe kategori sebagai vektor biner yang bernilai integer, 0 dan 1. Semua elemen akan bernilai 0 kecuali satu elemen yang bernilai 1, yaitu elemen yang memiliki nilai kategori tersebut.

Apa pertimbangannya memilih OHE pada feature **Subway Station**? yaitu karena tidak adanya hirarki pada data tersebut

Ordinal Encoding

Original Encoding	Ordinal Encoding
Poor	1
Good	2
Very Good	3
Excellent	4

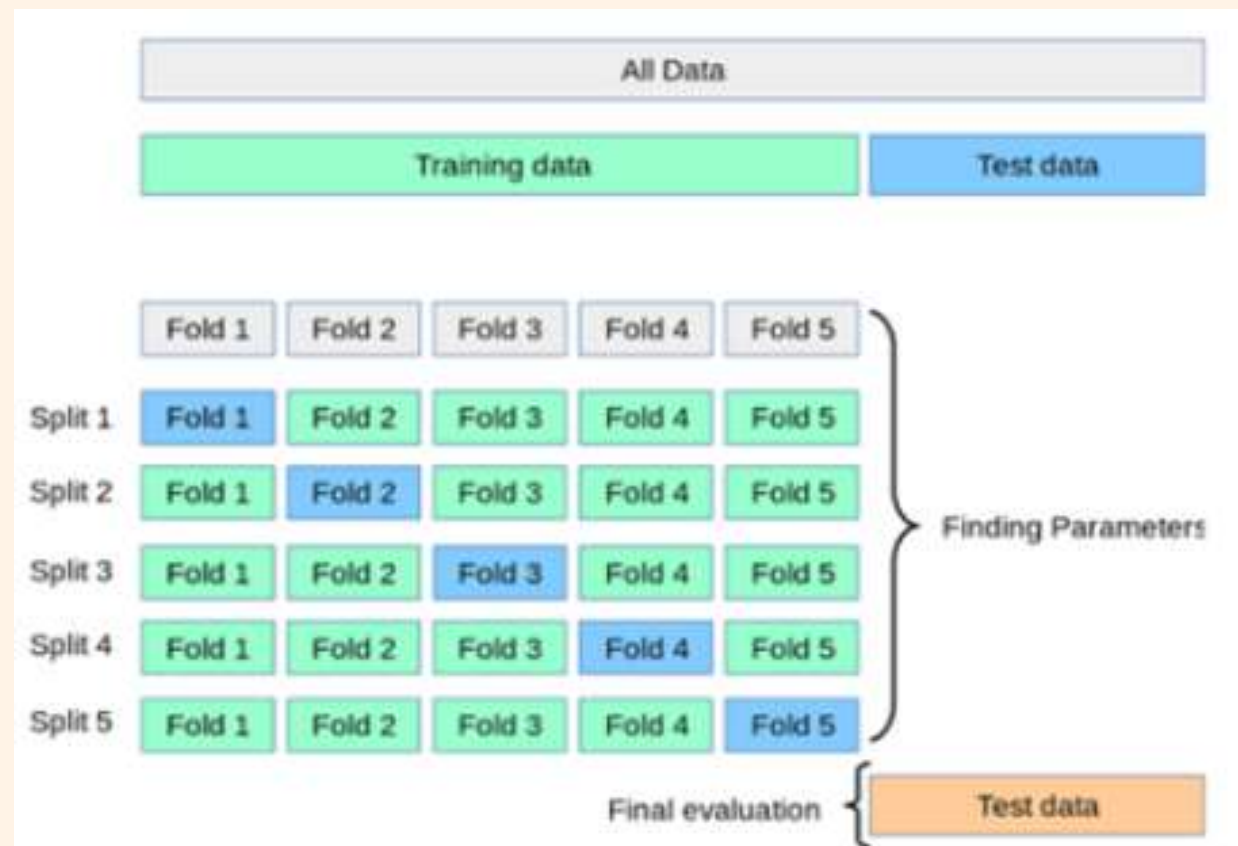
TimeToSubway dimasukan sebagai ordinal karena tentu saja semakin kecil durasi perjalanan menuju station, maka semakin baik.

Kandidat Algo

- KNN: adalah sebuah algoritma untuk melakukan klasifikasi terhadap objek dengan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Model ini diterapkan pada kasus yang sifatnya non parametrik.
- Decision Tree: adalah algoritma yang menggunakan seperangkat aturan untuk membuat keputusan dengan struktur seperti pohon yang memodelkan kemungkinan hasil, biaya sumber daya, utilitas dan kemungkinan konsekuensi atau resiko.
- Random Forest: Random Forest: adalah modifikasi dari metode bagging, tujuannya juga sama yaitu membuat prediksi menjadi lebih stabil dan memperkecil ragam, perbedaannya dalam random forest kita akan memilih secara acak kandidat fitur yang akan digunakan dalam setiap splitting tree.
- Gradient Boosting: adalah algoritma yang memberikan model prediksi dalam bentuk ansambel model prediksi yang lemah.
- Lasso: model ini menggunakan regularization. Dengan mendorong model yang sederhana dan jarang (parameter lebih sedikit). Model ini sangat cocok untuk model yang menunjukkan tingkat multikolinieritas tinggi, maka dari itu model ini saya pertimbangkan juga mengingat adanya kasus multikolinieritas pada tahap sebelumnya
- Ridge: adalah metode untuk memperkirakan koefisien model regresi berganda dalam skenario di mana variabel independen berkorelasi tinggi. Maka dari itu diharapkan model ini bisa menangani masalah multikolinieritas.
- Adaboost: merupakan salah satu teknik ensemble dengan menggunakan loss function fungsi exponential untuk memperbaiki tingkat akurasi dari prediksi yang dibuat.

Penjelasan Terkait Variabel

Crossval



Untuk mengetahui performa dari suatu model algoritma dengan melakukan percobaan sebanyak k kali

Menurut studi nilai 10 adalah yang paling sering digunakan, alasannya adalah akan menghasilkan trade off yang baik antara komputasional rendah dan bias yang minuman ketika melakukan estimasi model performance (machinelearningministry.com)

Random State

random state	0	1	2	3	---	
1	5	3	8	2		train dataset
2	3	8	2	1		
3	7	10	4	9		
4	1	9	3	3		
5	9	6	10	7	---	
6	6	7	5	6		
7	8	2	6	10		
8	4	1	7	8		
9	10	4	9	5		test dataset
10	2	5	1	4	---	

shuffled datasets

test size = 0.2

<https://medium.com/mllearning-ai/what-the-heck-is-random-state-24a7a8389f3d>

Power Transform by Yeo Johnson

$$\psi(\lambda, y) = \begin{cases} ((y + 1)^\lambda - 1) / \lambda & \text{if } \lambda \neq 0, y \geq 0 \\ \log(y + 1) & \text{if } \lambda = 0, y \geq 0 \\ -[(-y + 1)^{2-\lambda} - 1] / (2 - \lambda) & \text{if } \lambda \neq 2, y < 0 \\ -\log(-y + 1) & \text{if } \lambda = 2, y < 0 \end{cases}$$

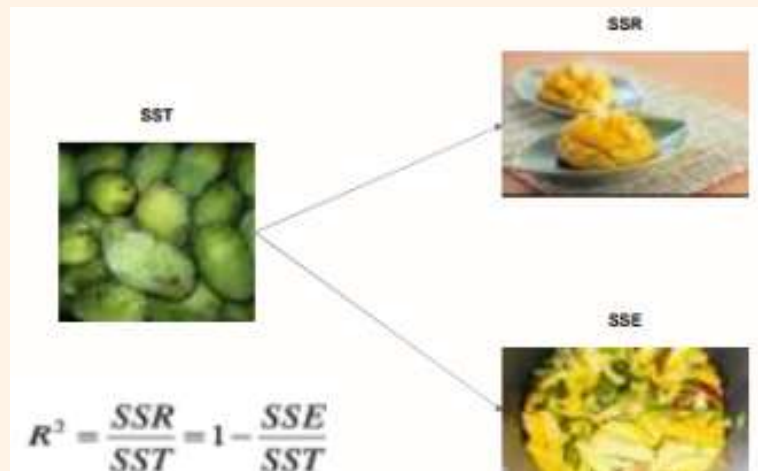
mengurangi skew

Evaluation Metrics

R2

R-Square dianalogikan ketika kita makan mangga, tentu kita ingin memakan daging buah sebanyak - banyaknya > seberapa baik merepresentasikan data

Higher is Better



MAE

karena adanya residual yang tidak stabil dan mudah diinterpretasikan

Lower is Better

MAPE

MAPE diperoleh dengan cara menghitung residual dalam bentuk persentase

Lower is Better

RMSE

besarnya tingkat kesalahan hasil prediksi

Lower is Better



Evaluation Metrics

	Mean_R2	Std_R2	Mean_RMSE	Std_RMSE	Mean_MAE	Std_MAE	Mean_MAPE	Std_MAPE
RandomForestRegressor(random_state=2022)	0.657456	0.038984	-46.661653	1.795464	-37.183647	1.593953	-0.180992	0.008392
DecisionTreeRegressor(random_state=2022)	0.657063	0.039657	-46.683138	1.781039	-37.196481	1.585094	-0.180888	0.007941
GradientBoostingRegressor(random_state=2022)	0.658936	0.038952	-46.558027	1.706605	-37.356741	1.393560	-0.181886	0.007858
KNeighborsRegressor()	0.595809	0.034882	-50.754993	1.983226	-38.590354	1.277831	-0.183291	0.008422
AdaBoostRegressor(random_state=2022)	0.581660	0.043432	-51.582414	1.388048	-43.728751	1.145309	-0.220943	0.009605
LinearRegression()	0.513696	0.042615	-55.637434	1.158588	-46.143582	1.047995	-0.219612	0.007773
Ridge(random_state=2022)	0.513763	0.042481	-55.634176	1.154968	-46.155877	1.039896	-0.219632	0.007727
Lasso(random_state=2022)	0.505262	0.040588	-56.130035	1.152599	-47.014882	1.048535	-0.224198	0.007224

Maka dari itu kita mendapatkan algoritma terbaik berdasarkan MAE,

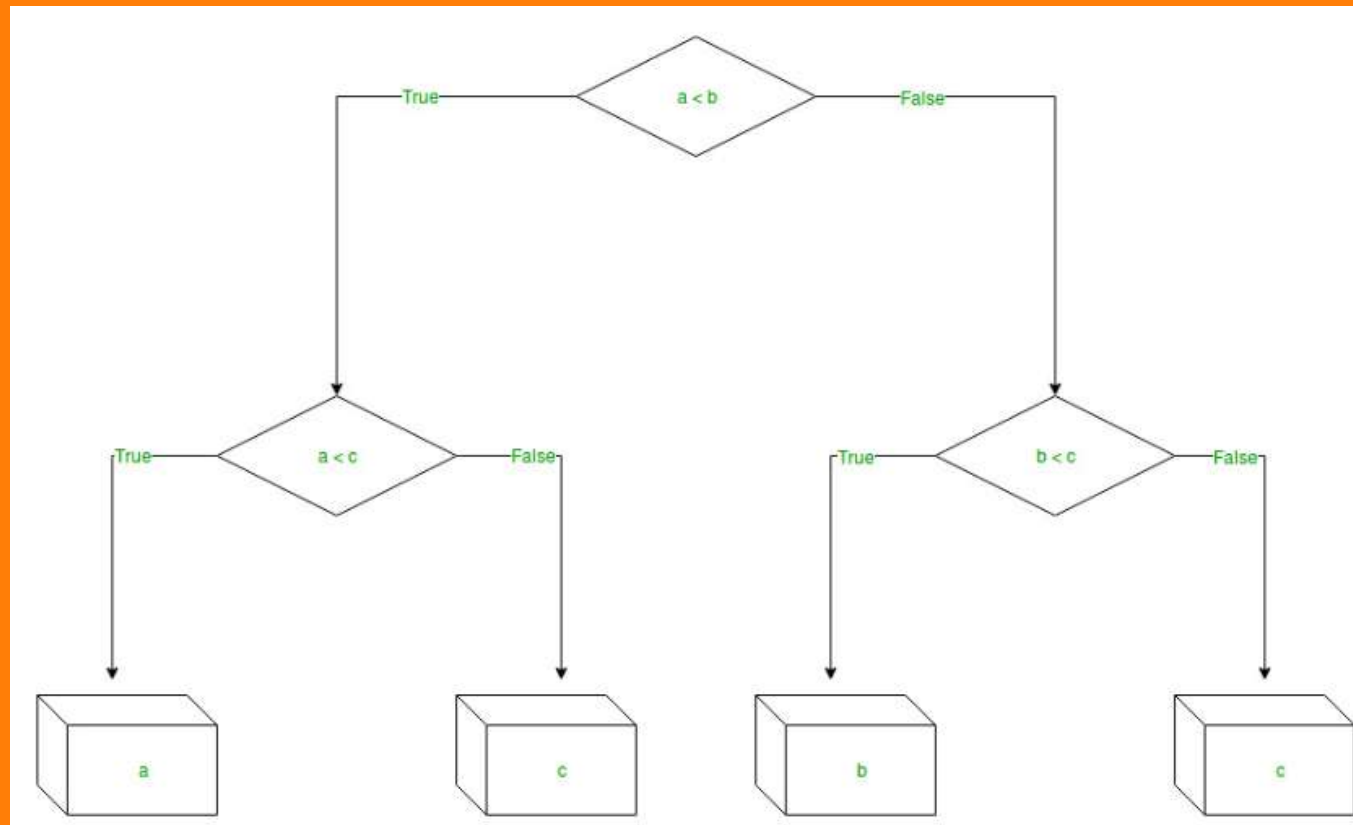
mengapa saya menggunakan MAE ? karena mudah diinterpretasikan dan ketika diagregasi pada dataset menggunakan rata-rata memiliki arti kesalahan rata-rata.

Maka dari itu kita mendapatkan 3 model terbaik pada sesi running kali ini, dengan error paling minimal yaitu sekitar 37% (lower is better). Selain itu jika kita lihat pada R2, ternyata juga selaras, ketiga algoritma tersebut mencapai nilai 60%, artinya bisa menjelaskan 60% keragaman pada model.



3 Metric Terbaik

Decision Tree



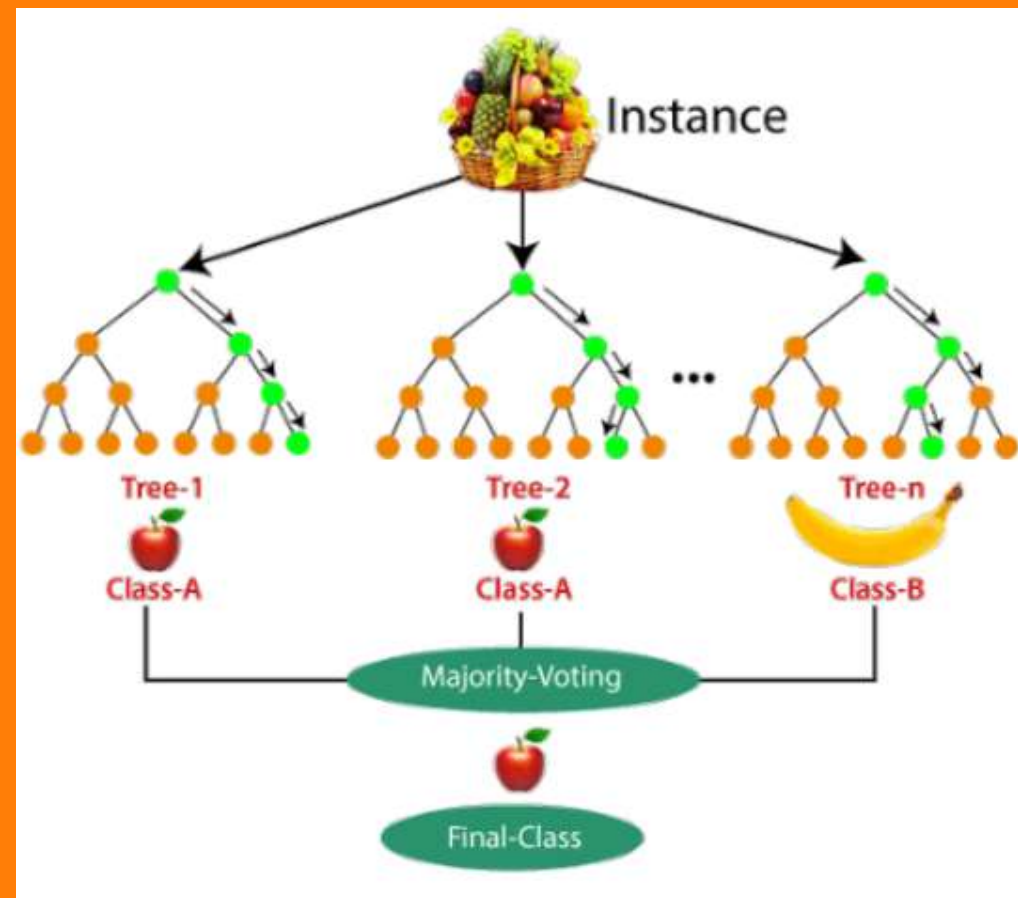
Decision Tree akan melakukan splitting dengan mempertimbangkan setiap feature dalam data training, lalu rata-rata dari balikan data training pada group tertentu akan dianggap sebagai prediksi group terkait. Kemudian akan diterapkan pada semua node dan cost untuk setiap kandidat yang akan di-splitting, tentu saja splitting dengan cost terendah akan dipilih.

Lalu kira-kira mengapa model ini bisa menjadi salah satu yang terbaik?

Karena model ini sifatnya non parameterik, jika kita membandingkan dengan algoritma seperti linear regresi, naive bayes, dll, sangat membutuhkan banyak asumsi agar model dapat bekerja secara efektif, sehingga jika ada permasalahan terhadap asumsi maupun distribusi, dapat terbantu dengan adanya model ini.

3 Metric Terbaik

Random Forest

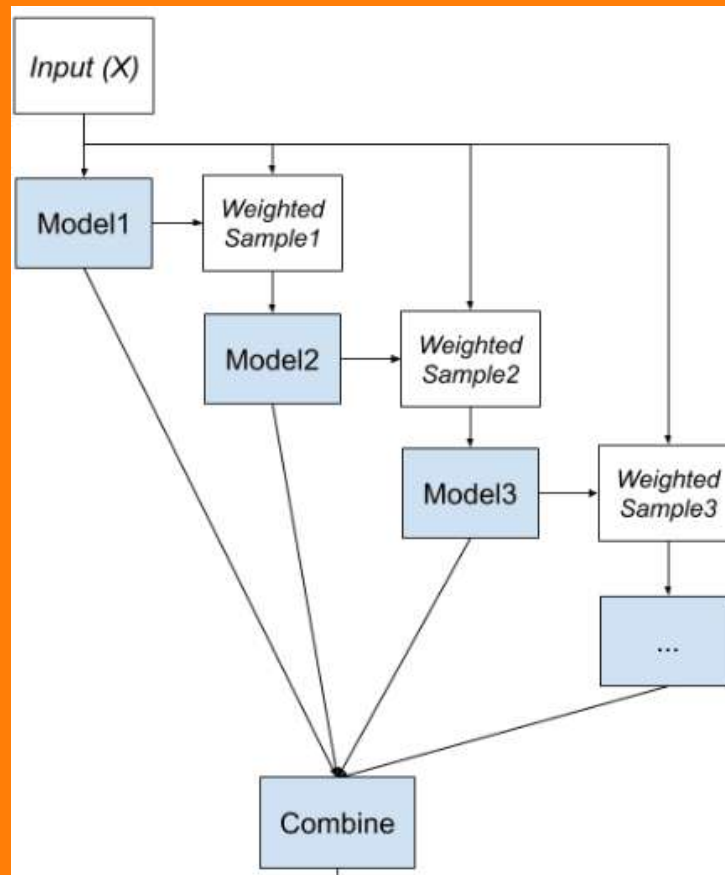


apakah random forest masih berhubungan dengan decision tree? Keduanya merupakan algoritma yang diterapkan pada machine learning. Berbeda, namun masih saling berkaitan. Mengapa demikian?

Random forest pada dasarnya merupakan kumpulan dari beberapa decision trees. “Pohon-pohon” tersebut saling berhubungan hingga akhirnya dapat menghasilkan prediksi yang lebih akurat dan stabil. Caranya adalah dengan mengambil rata-rata output dari “pohon-pohon” tersebut. Makin banyak decision tree yang digunakan, maka makin tinggi tingkat akurasi.

3 Metric Terbaik

Gradient Boosting



Gradient boosting dimulai dengan menghasilkan pohon klasifikasi awal dan terus menyesuaikan pohon baru melalui minimalisasi fungsi kerugian (Natekin dan Knoll, 2013).

Gradient Boosting adalah salah satu varian dari ensemble method yang mempelajari model yang lemah, kemudian dikombinasikan untuk mendapatkan performance yang lebih baik

Predict Test Set

Ketika dilakukan prediksi pada test set, performa tidak memiliki perbedaan signifikan dengan train set. Sehingga, dapat diartikan bahwa model tidak mengalami overfitting atau underfitting.

In all Estimators:

- `model.fit()` : fit training data. For supervised learning applications, this accepts two arguments: the data `X` and the labels `y` (e.g. `model.fit(X, y)`). For unsupervised learning applications, this accepts only a single argument, the data `X` (e.g. `model.fit(X)`).

In supervised estimators:

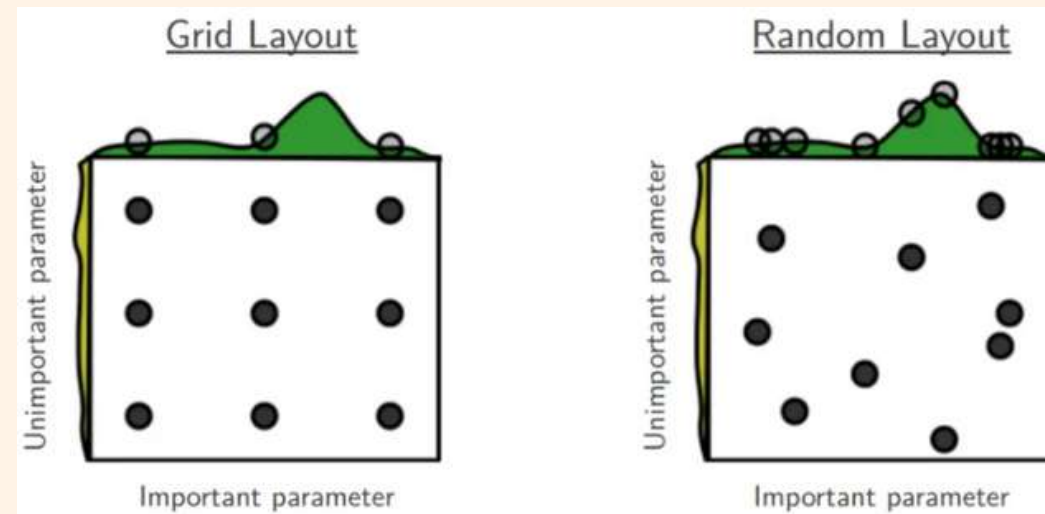
- `model.predict()` : given a trained model, predict the label of a new set of data. This method accepts one argument, the new data `X_new` (e.g. `model.predict(X_new)`), and returns the learned label for each object in the array.

	R2	RMSE	MAE	MAPE
RandomForest	0.635417	49.300234	39.289128	0.188788
DecisionTreeRegressor	0.634910	49.334505	39.376586	0.189582
GradientBoostingRegressor	0.634800	49.341907	39.471671	0.190584

Hyperparameter Tuning

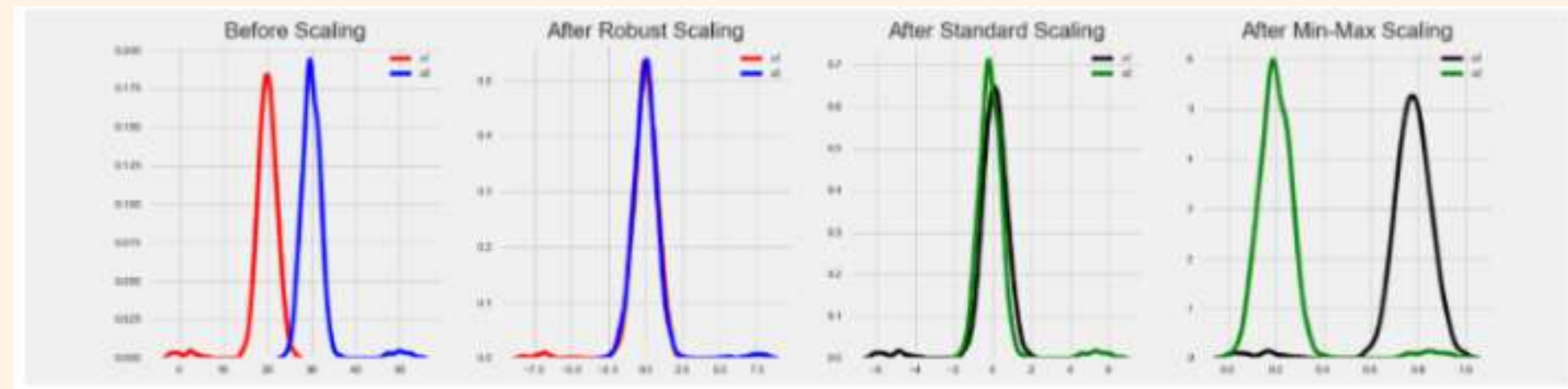
memastikan bahwa model machine learning memiliki kinerja yang baik dan memenuhi harapan. Ini membantu meningkatkan akurasi model machine learning

random search



Random Search: tidak setiap kombinasi dievaluasi, sehingga computational cost-nya pun menjadi lebih sedikit.

robust scaler



Ternyata ada yang berhasil dituning, bahwa error dapat berhasil diturunkan. Yaitu sekitar sekitar 5 - 15 USD per square feet

Before & After Tuning

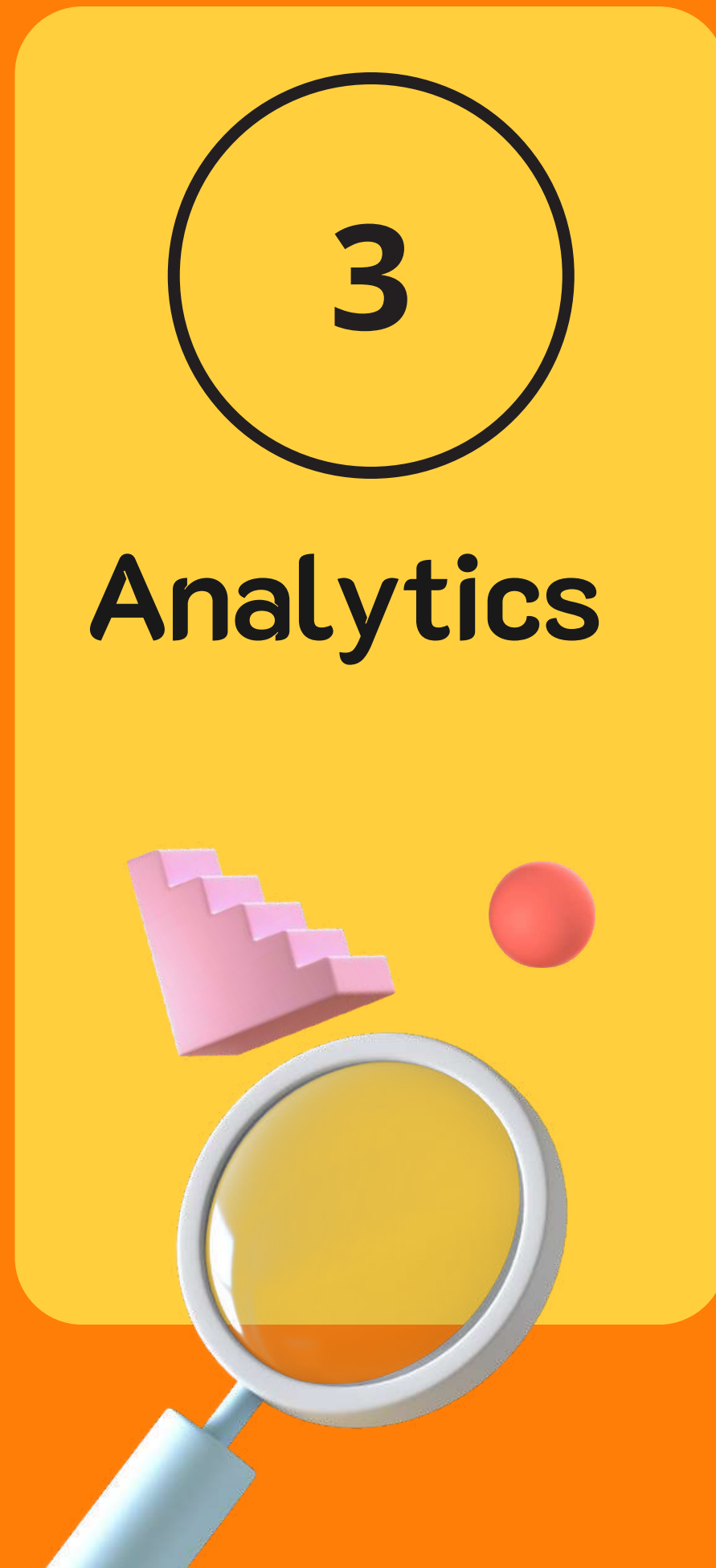
Before Tuning

	R2	RMSE	MAE	MAPE
RandomForest	0.635417	49.300234	39.289128	0.188788
DecisionTreeRegressor	0.634910	49.334505	39.376586	0.189582
GradientBoostingRegressor	0.634800	49.341907	39.471671	0.190584

After Tuning

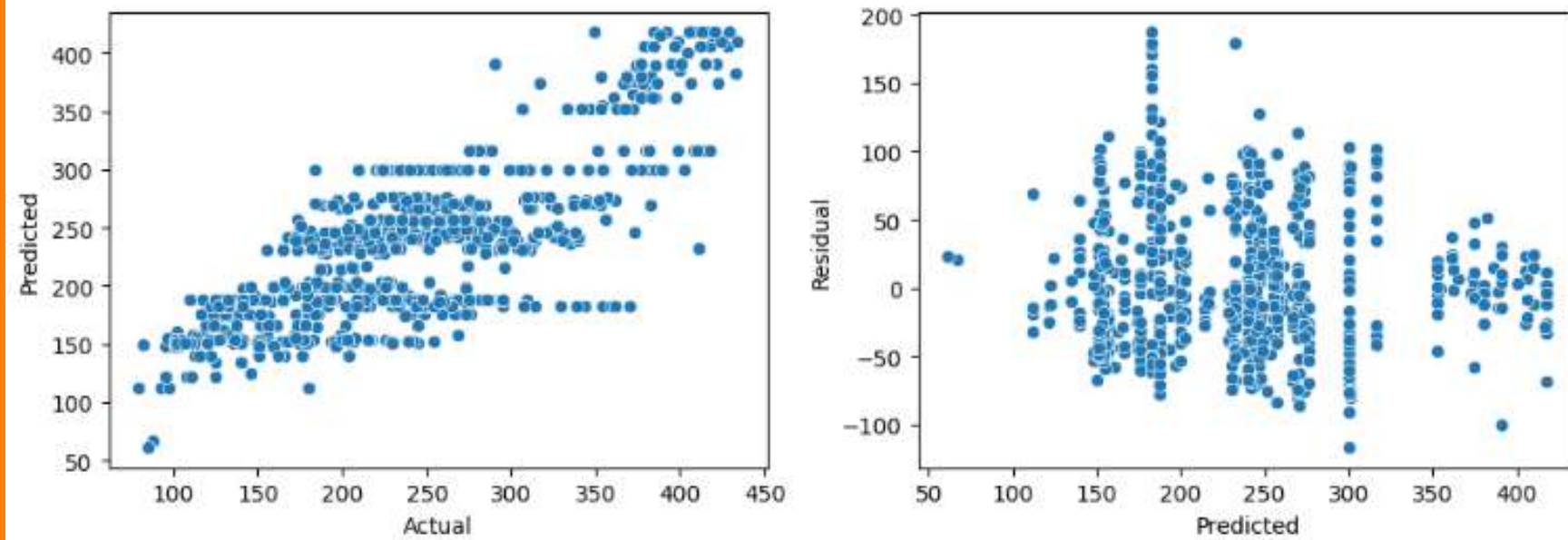
	R2	RMSE	MAE	MAPE
GradientBoostingRegressor	0.635247	49.311685	39.232211	0.188634
DecisionTreeRegressor	0.631005	49.597598	39.677377	0.192147
RandomForestRegressor	0.632080	49.525340	39.592631	0.192067





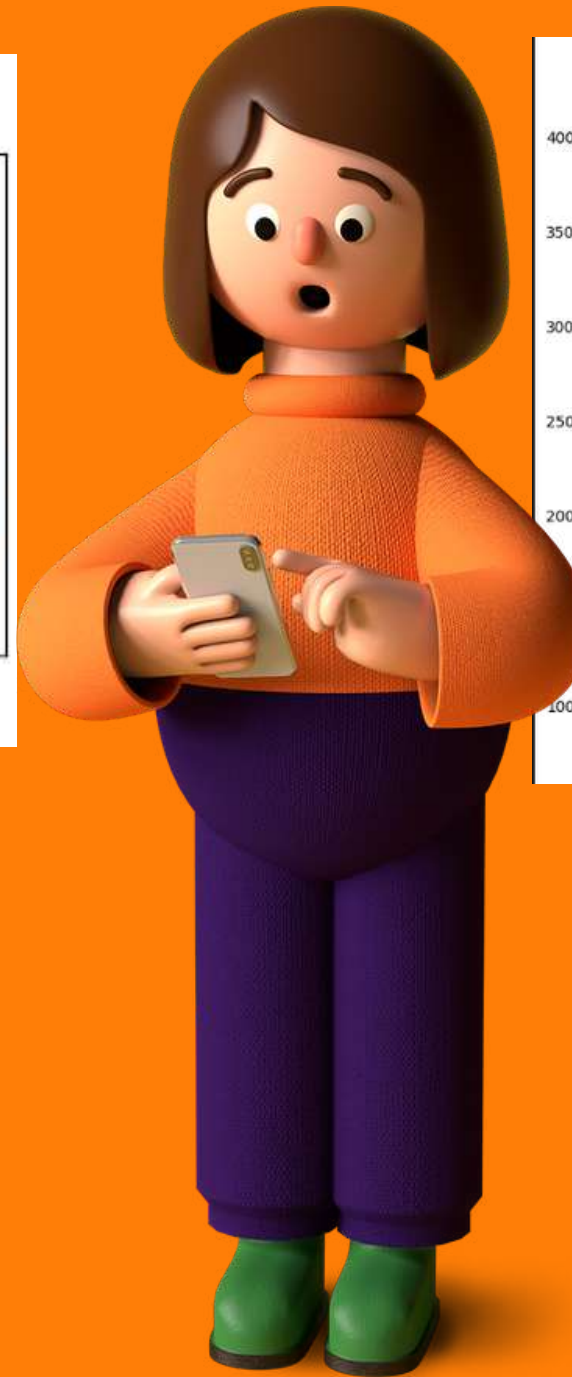
Analytics

Actual vs. Prediction Value

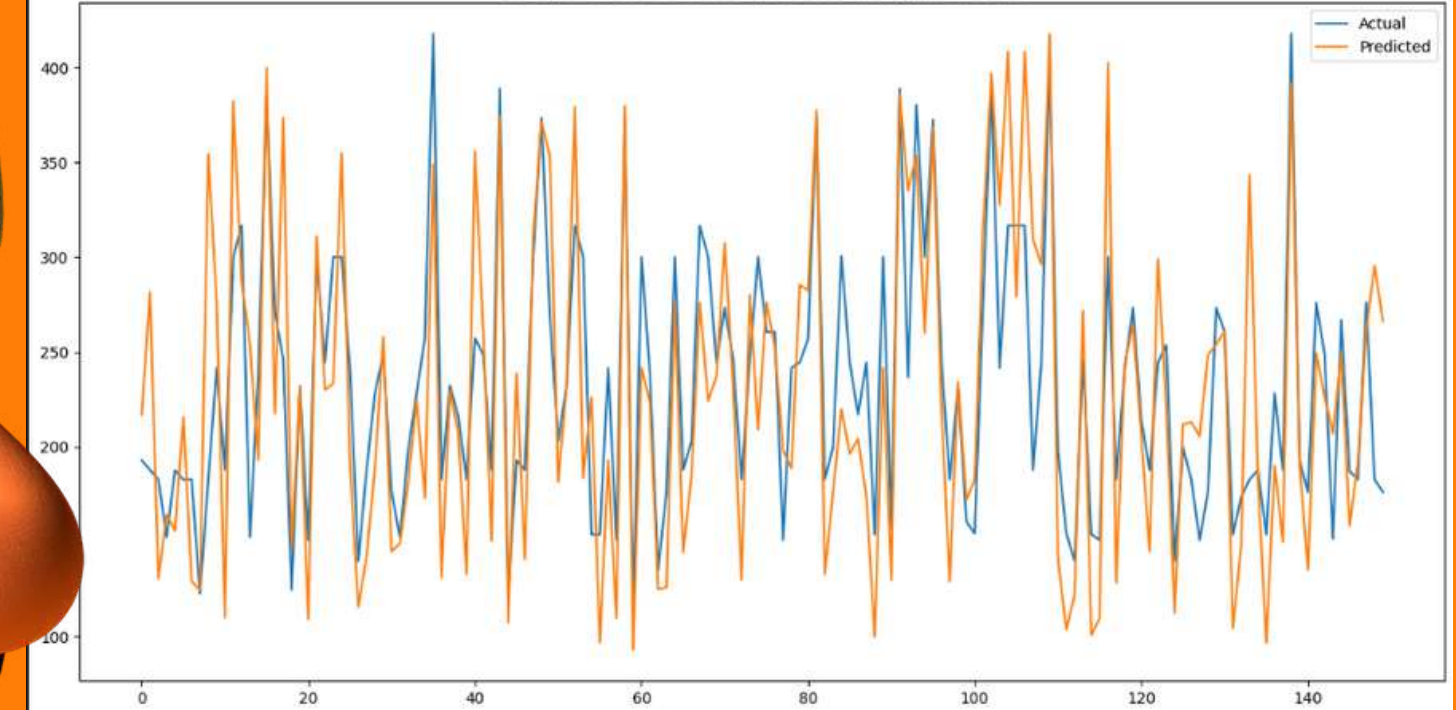


Jika kita lihat dari grafik actual vs prediction, prediksi sudah merapat pada satu titik, namun ada sedikit kecenderungan untuk menyebar, terutama pada nilai di atas 200. Hal ini menandakan masih adanya potensi bias pada prediksi yang sudah dibuat.

Lalu pada plot predicted vs residual, pada prediksi harga sekitar 175 memberikan hasil yang kurang baik

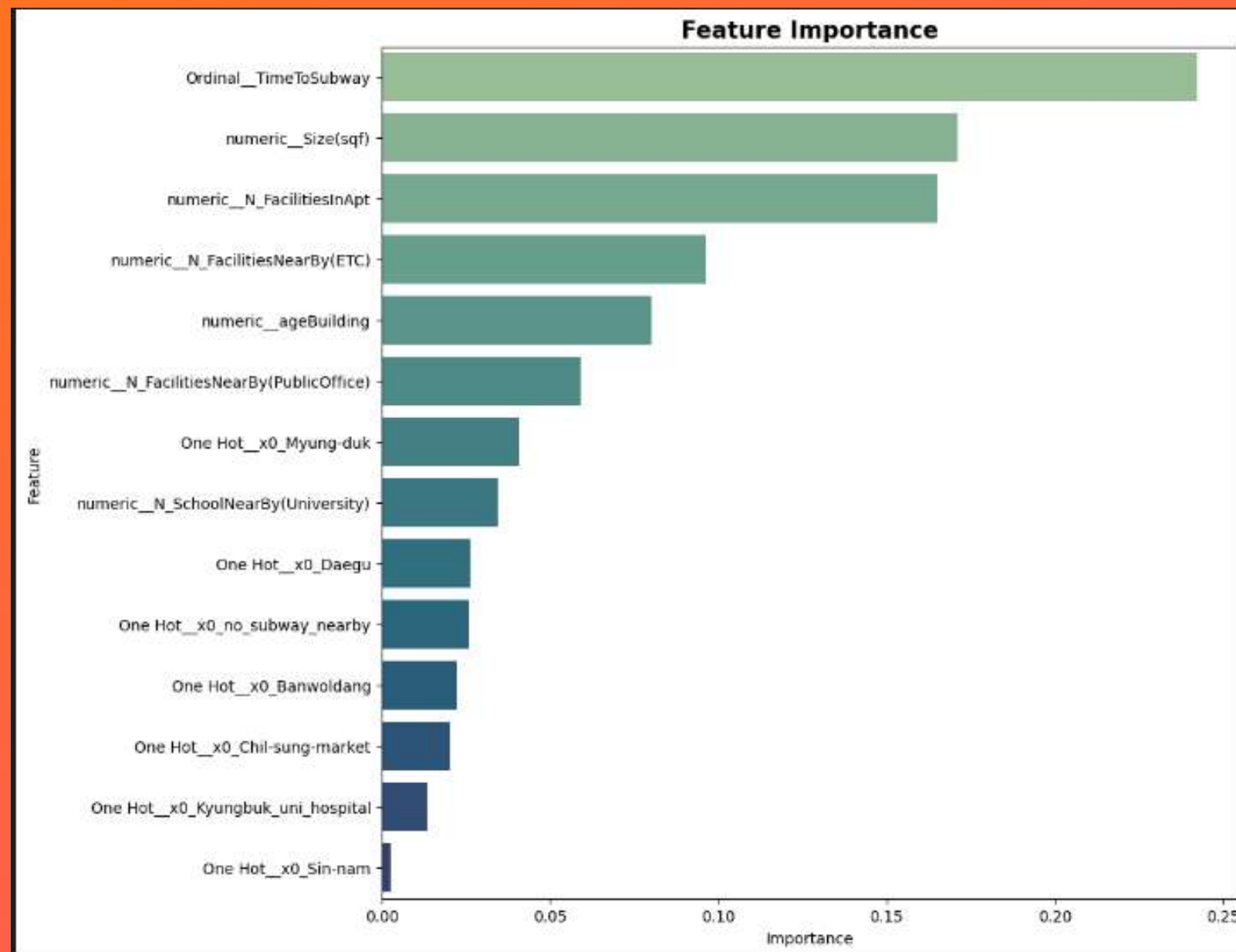


Compare Best Model predict dengan Actual



Pola actual dan predicted hampir sama

Feature Importance



Feature Importance: kontribusi masing-masing fitur terhadap prediksi model.

Hal ini cukup masuk akal, karena kemudahan akses transportasi juga menjadi nilai tambah ketika kita memilih tempat tinggal.

Lalu umur bangunan juga mungkin dapat berpengaruh pada harga, biasanya bangunan yang baru cenderung lebih kuat, bersih, dan biasanya secara design lebih modern (meskipun estetika sangat subjektif untuk diukur).

Yang terakhir yaitu Size, jika kita berkaca pada realitas, ukuran memang menjadi hal penting untuk mahal / murah nya sebuah tempat tinggal, bahkan di Indonesia sendiri sering menggunakan satuan per meter untuk mengkomunikasikan harga.

Lalu mengapa beberapa stasiun nilainya justru rendah? dugaan saya karena sebenarnya variabel tersebut sudah diwakilkan oleh time to subway, orang akan cenderung memperhatikan jarak/waktu ke stasiun terdekat daripada di mana stasiunnya, jika asumsinya seseorang memilih suatu apartment karena dekat dengan kantor, maka sebenarnya pertimbangan SubwayStation tidaklah terlalu penting, karena sehari-hari orang tersebut ke kantor dengan jalan kaki / kendaraan, bukan naik kereta.



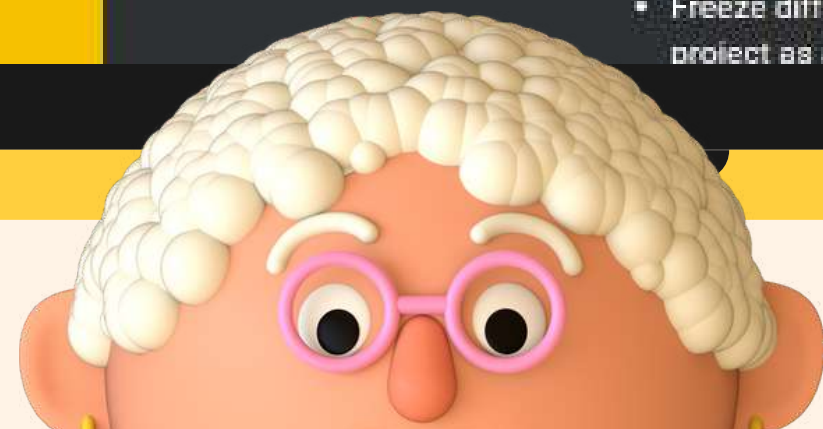
Features

- Compatible with Shap, Lime and ACV
- Uses shap backend to display results in a few lines of code
- Encoders objects and features dictionaries used for clear results
- Compatible with category_encoders & Sklearn ColumnTransformer
- Visualizations of global and local explainability
- Webapp to easily navigate from global to local
- Summarizes local explanation
- Offers several parameters in order to summarize in the most suitable way for your use case
- Exports your local summaries to a Pandas DataFrame
- Usable for Regression, Binary Classification or Multiclass
- Compatible with most of sklearn, lightgbm, catboost, xgboost models
- Relevant for exploration and also deployment (through an API or in Batch mode)
- Freeze different aspects of a data science project as a basis of an audit report

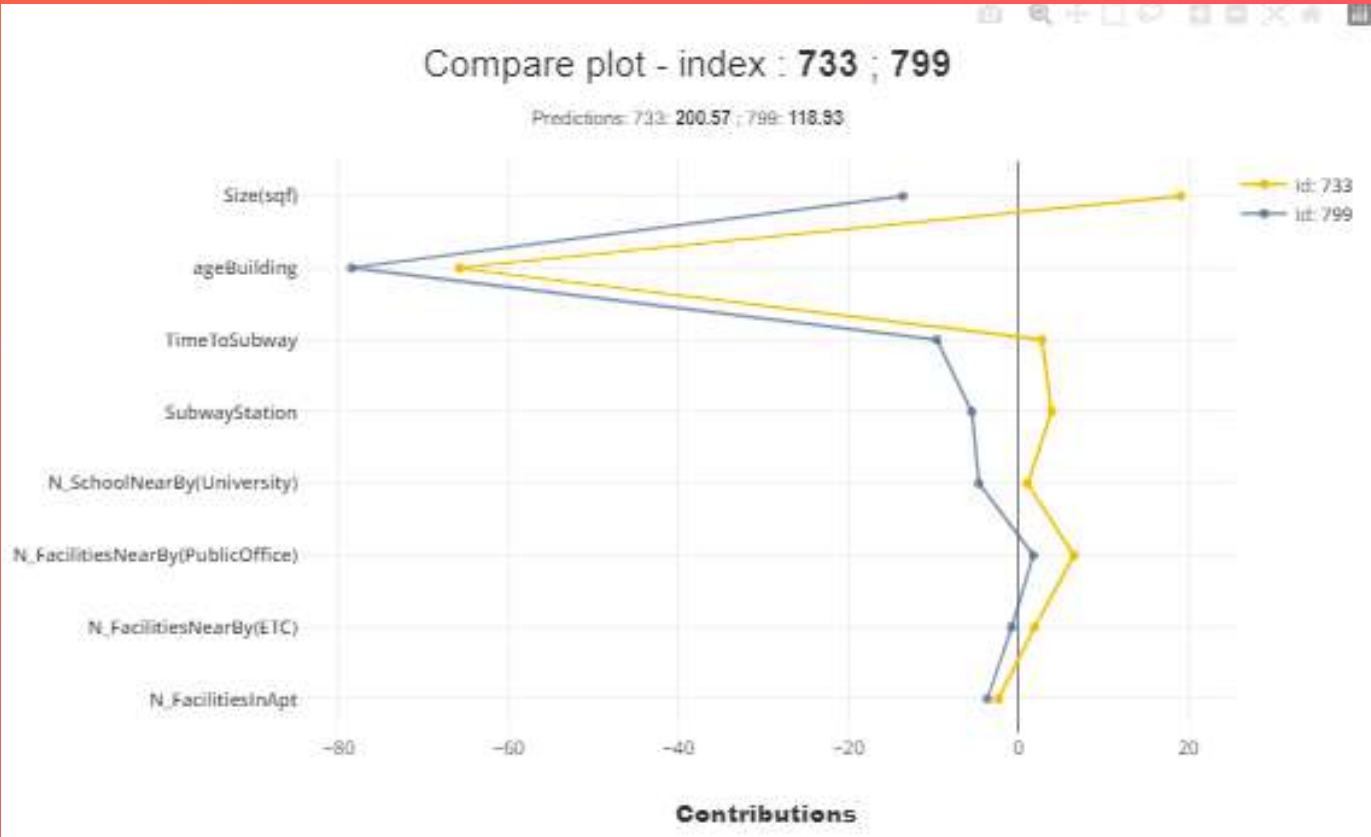
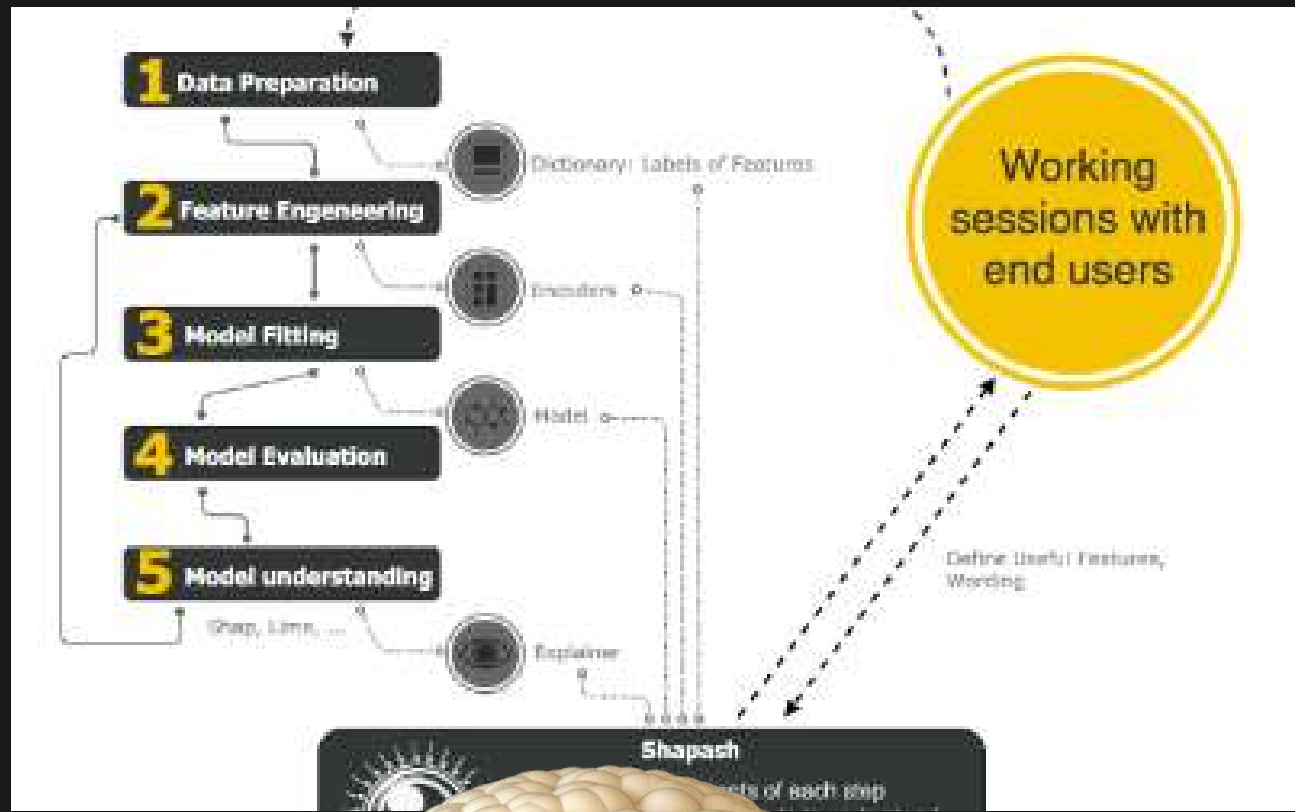
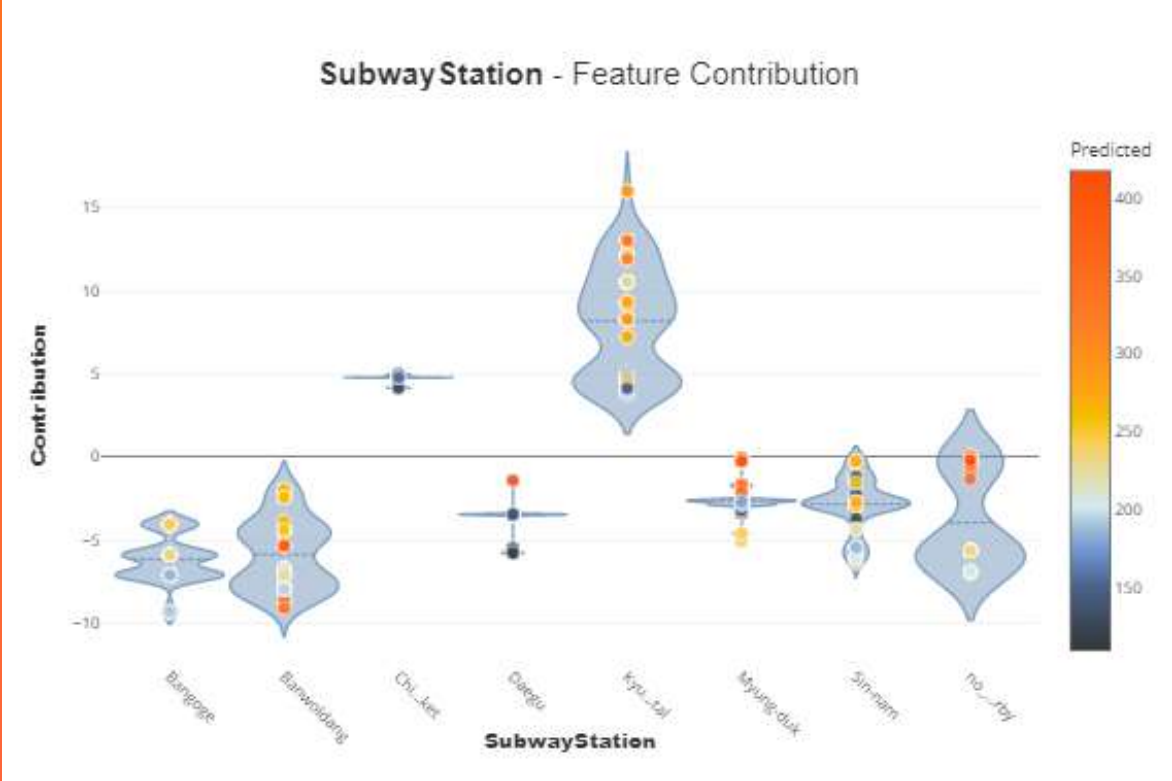


SHAPASH

SHAPASH adalah library Python yang bertujuan untuk membuat ML dapat ditafsirkan dan dimengerti oleh semua orang. Shapash menyediakan beberapa jenis visualisasi yang menampilkan label eksplisit yang dapat dipahami semua orang. End User juga dapat memahami keputusan yang diusulkan oleh model menggunakan ringkasan kriteria yang paling berpengaruh.



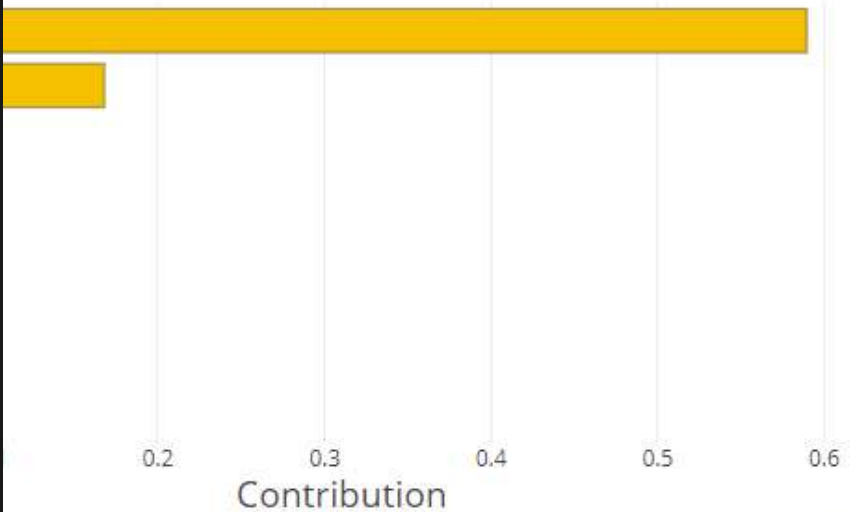
SHAPASH



Daegu Apartments

Regression Classification

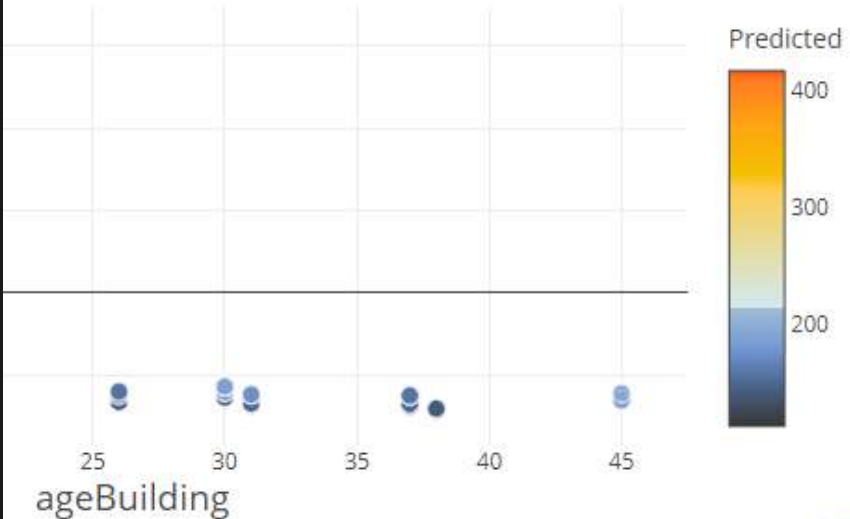
Features Importance



Dataset Dataset Filters True Values Vs Predicted Values

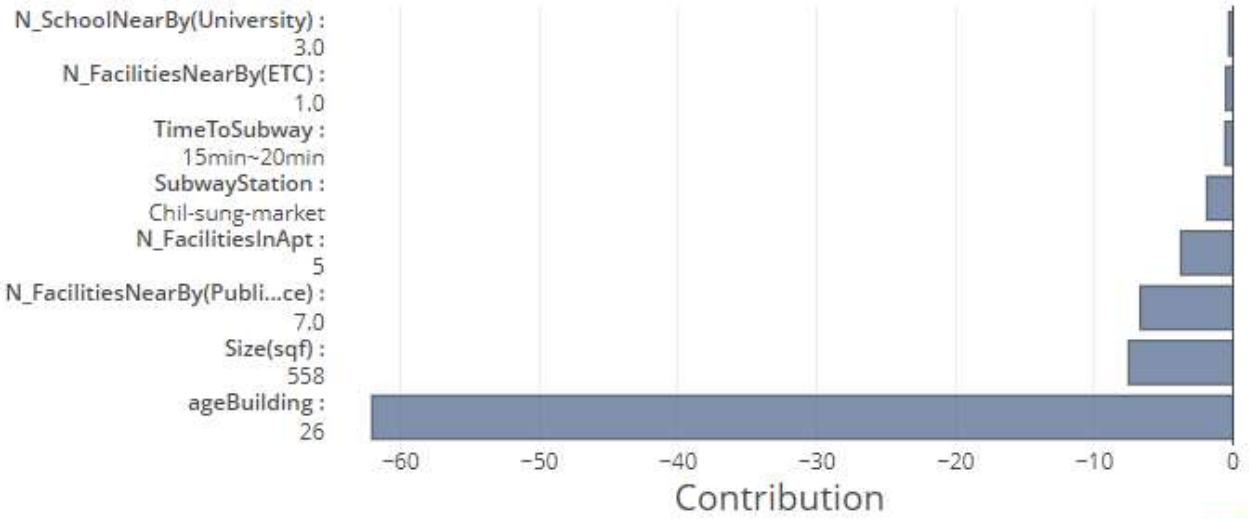
index	predi	TimeToSubway	SubwayStation	N_Facilities	N_Facilities	N_SchoolNear	N_Facilities
87	188.1	5min~10min	Bangoge	5	5	4	7
88	198.3	0-5min	Banwoldang	0	2	2	4
90	233.3	0-5min	Kyungbuk_uni_hospital	0	5	3	7
91	233.3	0-5min	Kyungbuk_uni_hospital	0	5	3	7
110	294.9	0-5min	Kyungbuk_uni_hospital	0	3	2	10
136	248.5	10min~15min	Kyungbuk_uni_hospital	1	2	2	5
138	187.9	15min~20min	Myung-duk	5	6	5	4
142	230.1	5min~10min	Bangoge	5	5	4	7
146	321.9	0-5min	Kyungbuk_uni_hospital	0	5	3	7
149	188.1	5min~10min	Bangoge	5	5	4	7

Building - Feature Contribution



Local Explanation - Id: 2

Predict: 150.1



Index

Threshold: 0

0 50 100

Features to display: 8

1 4

Contributions to display:

☒ Positive ☒ Negative

Feature(s) to mask:

Select...

Note: di sini saya menggunakan Random Forest

Kesimpulan

- Dataset ini cenderung lebih optimal dengan algoritma yang tree based serta boosting. Tree-based model ini cocok untuk dataset dengan berbagai tipe data, kita tahu bahwa dataset Daegu Apartments terdapat beberapa tipe data, yaitu numerikal dan kategorikal. Kemudian tree based model juga cocok untuk data yang sifatnya tidak terdistribusi normal. Selain itu Decision Tree juga dapat diterapkan pada data yang mengalami masalah non linieritas.
- Jika dilihat dari nilai RMSE yang dihasilkan oleh model setelah dilakukan hyperparameter tuning, yaitu sebesar 48 USD per square feet, kita dapat menyimpulkan bahwa perkiraan meleset kurang lebih sebesar 48 USD per square feet. Tetapi, tidak menutup kemungkinan juga prediksinya meleset lebih jauh karena bias yang dihasilkan model masih cukup tinggi pada harga prediksi yang semakin tinggi, seperti visualiasi yang telah ditunjukkan pada asumsi asumsi sebelumnya. Bisa dibuktikan juga pada pengujian model terhadap data test memiliki nilai yang meleset yang paling tinggi diatas 186 USD per square feet(diff). Hal ini terjadi karna adanya bias yang cukup tinggi pada model.
- Untuk hasil MAPE di bawah 20% dapat tergolong baik
- Hasil dari Rsquare yang di atas 60% sudah cukup baik. R square adalah seberapa baik regresi merepresentasikan data.

MAPE	Judgement of forecast accuracy
Less than 10%	Highly accurate
11% to 20%	Good forecast
21% to 50%	Reasonable forecast
51% to more	Inaccurate forecast
Source: Data from Lawrence and Klimberg (2016).	
Scale of judgment of forecast accuracy, Lewis scale-1982	

R Squared value	Interpretation
0.75 - 1	Significant amount of variance explained
0.5 - 0.75	Good amount of variance explained
0.25 - 0.5	Small amount of variance explained
0 - 0.25	Little to no variance explained

Manfaat Bisnis

Apa manfaatnya?



Berdasarkan informasi yang saya baca dari https://hingemarketing.com/blog/story/cost_and_benefits_of_market_research, dengan Traditional Valuation, market researchers butuh waktu 2 - 8 minggu untuk melakukan riset harga terbaik dari satu apartment unit.

Sedangkan dengan Advance Valuation (machine learning model), kita dapat melakukan prediksi 1 unit apartment price kurang dari satu hari.

Anggaplah biaya market riset dalam 4 minggu adalah 100 USD per hari, jika dalam 4 minggu(dikurangi weekend / hari istirahat) maka total biayanya adalah 2000 USD. Jika dibandingkan dengan menggunakan ML yang hanya kurang dari 1 hari, maka perbandingannya adalah 100 USD : 2000 USD, atau **lebih hemat 95 %** (di luar estimasi biaya pengembangan ML).

Selain itu dari sisi Buyer, dapat mereduksi resiko pembelian apartments dalam harga yang terlalu mahal

Dari sisi Penjual, akan bermanfaat dalam menentukan harga terbaik, supaya tidak terlalu mahal ataupun terlalu murah (sweet spot).



Limitasi

Ada batasannya?



1. Karena adanya penghapusan terhadap outlier, maka akan ada limit pada batas maksimal harga dan size yang dimasukan, untuk harga maximal yang bisa dimasukan adalah 510747.0, sedangkan size maximal yang bisa dimasukan adalah 1906.5
2. Pada fitur ageBuilding tidak dinamis / perlu diupdate kembali setiap ada penambahan tahun.
3. Karena kita menggunakan tree-based model, akan ada beberapa kekurangan,
 - pertama, jika datanya terlalu besar (big data), maka akan terlalu banyak node yang terbuat, sehingga model menjadi terlalu kompleks dan overfitting.
 - Selain itu terkait reusability-nya, karena perubahan kecil pada data (misal suatu saat ingin memasukan dataset yang serupa) dapat menyebabkan perubahan struktur secara signifikan (tree tidak stabil)
4. Masih sedikit bias seperti yang sudah dijelaskan pada Jika kita lihat dari grafik actual vs prediction.



Rekomendasi

Terhadap Model



- Mempertimbangkan juga dengan kota besar yang lain seperti Seoul dan Busan, karena wilayah geografis juga terkadang berpengaruh terhadap harga properti.
- Pergunakan Grid Search, karena biasanya Grid Search juga dapat memiliki dampak pada performa secara lebih baik. Namun tentu ada trade off-nya, yaitu computational cost-nya, berarti membutuhkan device dengan spesifikasi yang baik.
- House Price Bubble, terjadi ketika suatu harga properti naik secara signifikan, sementara kapabilitas orang untuk membeli properti rendah. Menurut Roberts, Lawrence (2008), agar pasar otomatis bisa menyadari adanya price bubble disarankan perubahan cara penilaian yang sudah ada yang menggunakan teknik sales comparison approach menjadi teknik income approach. Sales comparison approach didasarkan pada harga bangunan yang dianggap relative sama pada transaksi jual beli terakhir sehingga kenaikan harga suatu rumah memicu kenaikan harga rumah lainnya. Kemudian kita juga bisa mempertimbangkan income / penghasilan rata rata pada tiap daerah.
- Gunakan algoritma lainnya, seperti deep learning, Recursive Neural Network, XG Boost.dll.
- Deploy dan buat GUI
- Jika konteks prediksi harga housing di Korea, akan lebih menarik bila dikontekstualisasikan dengan Krisis Korea tahun 1997 - 1998.

Rekomendasi

Terhadap Bisnis



Latest Renovation



Berhubung umur bangunan ternyata menjadi salah satu fitur yang penting, hal ini dapat dijadikan pertimbangan bagi pengembang /penjual apartment, namun umur bangunan tentu saja sifatnya fix / tidak bisa diubah, kecuali dirobohkan lalu dibangun lagi. Maka dari itu, bisa dilakukan renovasi secara teknis, kita juga bisa menambahkan fitur 'Latest Renovation' / kapan terakhir bangunan tersebut diperbaharui. Trade off-nya adalah biaya tambahan bagi pengembang, karena tentu saja melakukan renovasi tidaklah murah. Lalu apa solusi lainnya? kita bisa menggunakan strategi marketing yang lain, yaitu dengan lebih menekankan pada kelebihan lainnya (misalnya fasilitas, ukuran, ataupun transportasi terdekat), sehingga customer secara psikologis tidak terlalu memikirkan kapan apartemen tersebut dibangun, namun lebih tertarik kepada fitur yang sudah ditekankan oleh marketing.

Sharing Profit



Dengan adanya machine learning, maka proses riset terhadap harga tentu menjadi lebih cepat, maka kita bisa bekerja sama dengan aplikasi jual beli apartment (seperti travelio). Misalnya dengan memberikan rekomendasi harga pada pemilik apartment yang ingin menjual unit-nya. Contoh proses bisnisnya secara singkat, seller input data apartment/ spesifikasi apartment, kemudian aplikasi akan memunculkan rekomendasi harga jual. Sehingga seller pun tidak bingung dalam menentukan harga, tidak terlalu mahal dan tidak terlalu murah. Ke depan bisa dievaluasi, apakah dengan menerapkan model / rekomendasi tersebut traffic penjualannya semakin bagus? apakah akan lebih cepat terjual? Jika iya, berarti model tersebut berguna bagi bisnis. Maka hubungan antara Model Developer dan Aplikasi Jual Beli Properti bisa menjadi saling menguntungkan (misalnya dalam bentuk sharing profit)

CONNECT WITH ME

thankyou



Email

andisetianto123@gmail.com



Git Hub

github.com/andsetiantowork